

unfolding Color Code の誤り耐性 その 2

前回に引き続き、Color Code の unfolding 操作について誤り耐性を調べてみた。特に unfolding する際の折り目の部分のエラー検出に関して詳しく解析した。

1. 折り目付近の誤り耐性

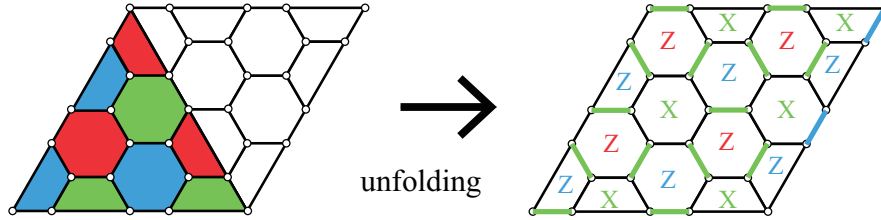


Fig. 1

前回までの unfolding 操作を Fig.1 に示す。前回の資料 (unfolding_color_code_2.pdf) ではあまり折り目の部分について議論しなかったが、よくよく考えてみると折り目の部分の誤り検出は思った以上に難しい。ここではそれを示す。

まず、最初 Color Code から始まったとき、左上部分には $|0\rangle$ 初期化されている qubit が用意してある。つまり、それらの qubit は 1-weight Z stabilizer でスタビライズされている。そのことを、水色を Z stabilizer として Fig.2(a) に示す。このとき、境界だけに注目すると、Fig.2(b) に示すように、red face Z stabilizer と付近の 1-weight Z stabilizer で六角形の Z stabilizer が構成できる。これより、unfolding 操作の折り目をまたがる 6-weight Z stabilizer によって折り目付近の X エラーを検出できる。

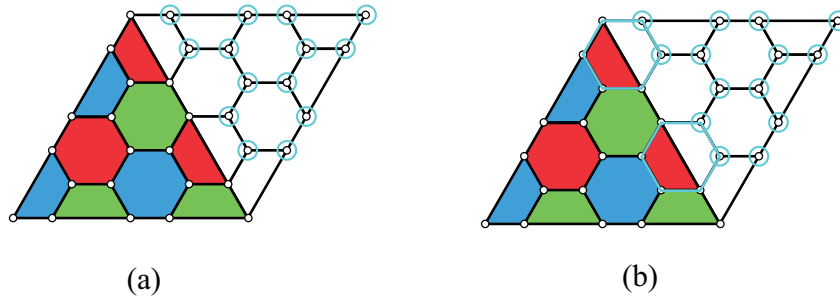


Fig. 2

しかし、Fig.3 に示すような破線の 6-weight Z stabilizer は undeterministic である。そのため、Fig.2(b) の上の紫色の六角形の Z stabilizer で検出されたエラーは Fig.3 の qubit 1,2,3,4 のどこで起きたのが全くわからない。そのようなことから、前回提示した unfolding のプロトコルは折り目付近の X エラーが取り除けない。

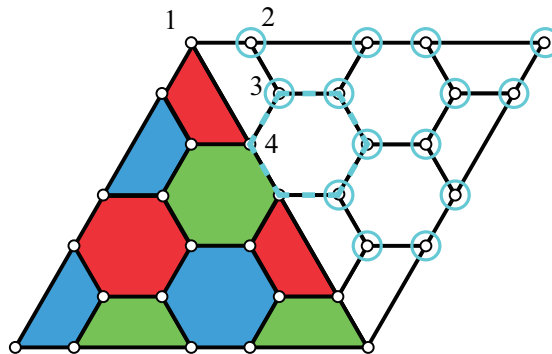


Fig. 3

また、前回のプロトコルだと折り目付近の Z エラーも取り除けないことに気がつく。しかしこれは、折り目付近の 2 つの qubit を Bell 状態にすることによって解決する。それを紫色を X stabilizer として Fig.4 に示す。ただし、このようにしても上記の X エラーを取り除けない問題は残る。このとき、Bell 状態の qubit の Z エラーも取り除ける。

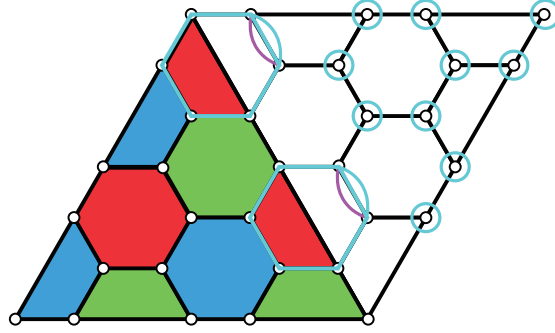


Fig. 4

ということで現状のプロトコルは修正が必要である。参考までに Fig.5 に試みた構造の残骸を載せておく。Fig.5 に示すどの構造をとっても (stabilizer 自体が成り立っていないものもある) 上記と似たような問題が発生した。

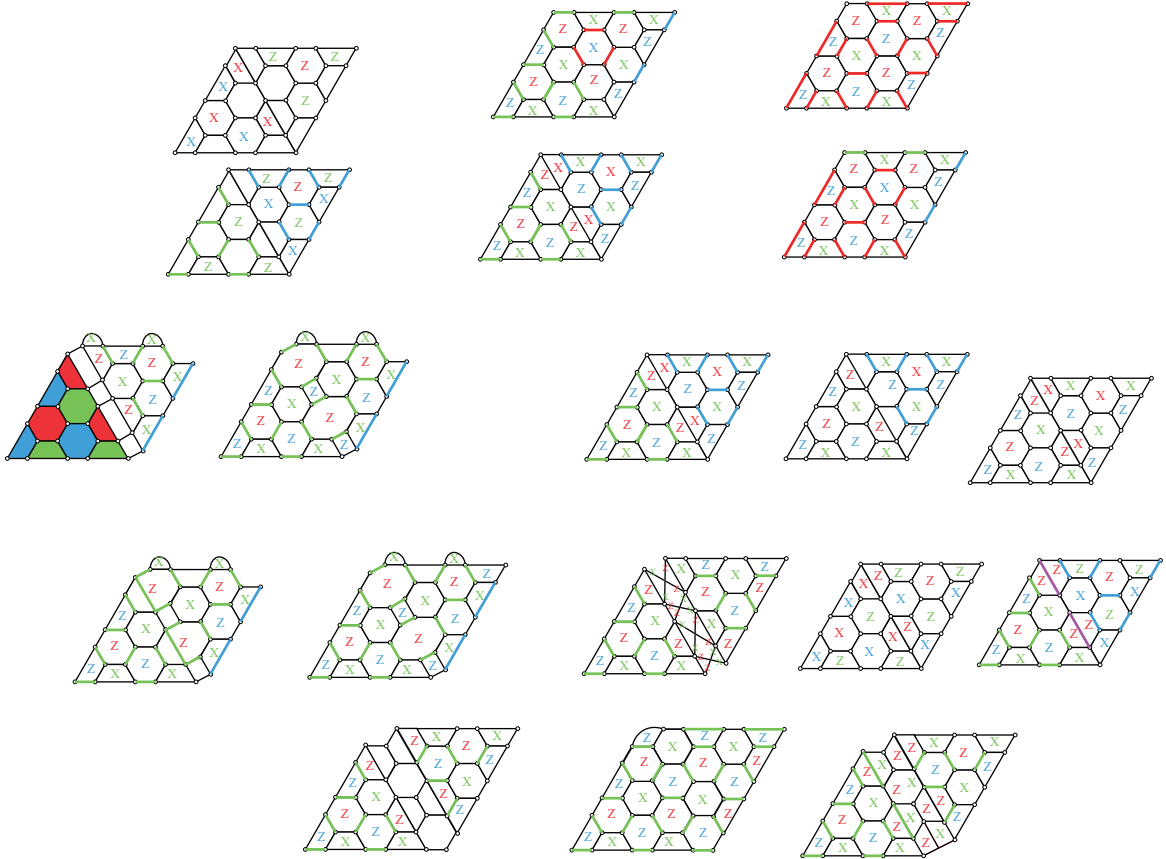


Fig. 5

ということで上記の問題の解決策になりそうなものを思いついたのでそれを以下で述べる。

2. Floquet Code

解決策として Floquet Code を用いるやり方を思いついたので、まずトーラス上での Floquet Code を使った Color Code の unfolding を考える。測定の種類や順番は Fig.6 示す通りである。ここで、Fig.6 にはシンドローム測定するスタビライザーしか示していない。実際には Fig.6 はもっと多く示していないスタビライザーが存在する。

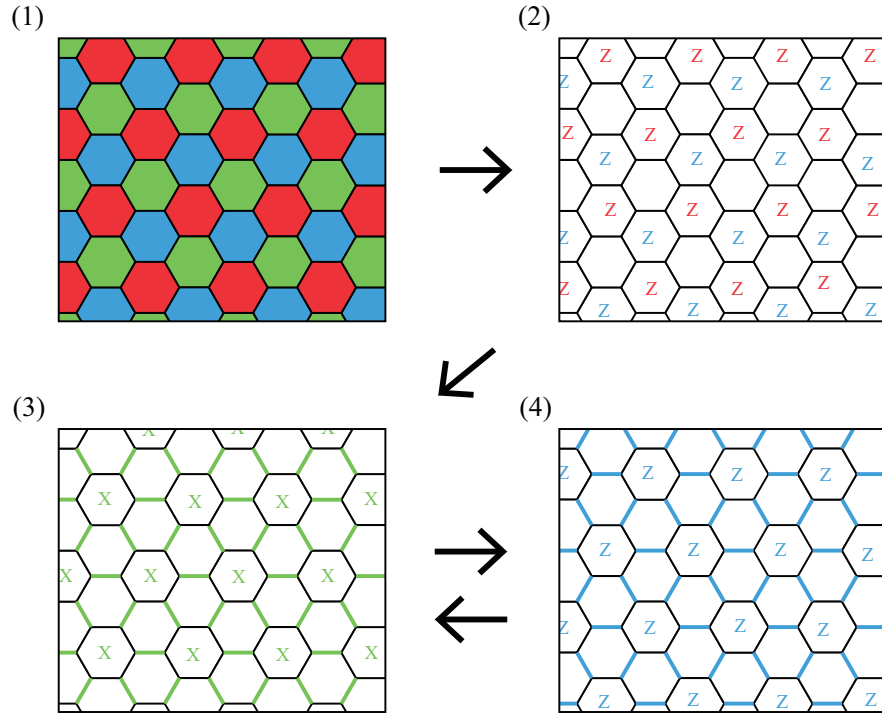


Fig. 6

ここから先、(1),(2),(3),(4) は Fig.6 の (1),(2),(3),(4) を表す。